

Foundations Project Background Reports

located at <http://bridges.state.mn.us/bestprac/reports.pdf>

Thesaurus Report

The Foundations Project uses and recommends a thesaurus, or controlled vocabulary, to aid users in finding information on Minnesota's state agency websites. It is located at: <http://bridges.state.mn.us/servlet/lexico>. The Legislative Indexing Vocabulary (LIV-MN), was chosen as the primary thesaurus for these reasons:

- **Vocabulary**
Devised and maintained by the Library of Congress and Congressional Research Service, LIV is used by the Federal government for statutes and thus is a good match for state government documentation. Another attribute of this vocabulary is its use of natural language and common terminology; non-librarians find it easy to work with.
- **Speed and Accuracy**
Through its unambiguous, clear language and lack of different synonyms or scattering of concepts, this vocabulary is easy to access and apply. Using a single controlled vocabulary eliminates syntax problems such as false hits or too many search results. Scope notes, cross references and related terms are also provided.
- **Flexibility**
It is possible to search both phrases and key words, ensuring that the most accurate term will be found.
- **Editing Capability with Lexico Software**
LIV is loaded on Lexico software, facilitating editing and the addition of new Minnesota-specific terms.

Dublin Core Report

Description

Dublin Core is a metadata schema that uses descriptors that resemble traditional tags used to create machine readable cataloging records; however, it was designed specifically to recognize, identify, locate, describe and retrieve information on the Internet. Dublin Core uses a simpler element set than MARC tags; Its descriptors are inserted into the HTML header portion of a Web page's source code and enhance the accuracy and precision of searches

The fifteen descriptors or elements of Dublin Core are in three basic categories:

Content Elements:

Title, Subject, Description, Source, Language, Relation, Coverage

Intellectual Property Elements:

Creator, Publisher, Contributor, Rights

Instantiation Elements:

Date, Type, Format, Identifier

Each of these descriptors can be extended by using explanatory sub-elements or type qualifiers. They're both optional and repeatable, making for a great deal of flexibility.

The characteristics of Dublin Core include:

- Simplicity - Dublin Core's set of elements is small and easy to use, especially for those not trained in library cataloging. With its limited element set, website developers and authors can insert Dublin Core metadata into source code easily and simply.
- Semantics - Dublin Core supports a common element set that is understood and supported across fields.
- International Scope - Dublin Core was originally developed in English, but now versions have been created in many other languages.
- Extensibility - Dublin Core's elements, while maintaining simplicity, may be extended and adapted for the needs of additional resource discovery in different research communities.

History

Dublin Core was developed at a series of six workshops that took place from 1995 to 1998. The name comes from the location of one of the founding organizations: OCLC (originally Online Library Computer Center) in Dublin, Ohio. The developers were an international group of people from both

the library/museum world and the computer, network and text encoding industry.

Each workshop used an international, multidisciplinary approach and focus groups to define and refine Dublin Core, direct further research and address problems. The first workshop developed thirteen core elements, the second created a proposed syntax and the Warwick Framework, the third expanded the element set to fifteen and the fourth devised formalized qualifiers and HTML-related expressions.

Much literature has resulted from these workshops. Considered in clusters, it includes proceedings and reports from the workshops; information about developing standards, mapping Dublin Core elements to other metadata systems as well as refining and qualifying it; and the implementation of Dublin Core into various projects. Research contributing to each facet of this literature is ongoing.

Discussion

Dublin Core's very simplicity makes it invaluable for many projects that bridge the research community. Its extensibility and repeatability allow for more elaboration when necessary. Dublin Core's combination of simplicity and complexity allows it to be easily installed in Web pages, even by non-specialists. In turn, Dublin Core makes resource discovery more accurate and expedient.

GILS Report

Description

The Government Information Location Service, or Global Information Locator Service (GILS) is a metadata schema designed to make it easier for people to locate, obtain and use online government information. It is intended as a tool for incorporation into government agencies' Web sites, standardizing information to make it more manageable and accessible.

GILS records may include many fields. Here are some examples:

- Title
- Local subject
- Abstract
- Purpose
- Available linkage type
- Agency program
- Access constraints
- Use constraints
- Cross reference
- Control identifier
- Date of last modification
- Record source
- Originator
- Thesaurus
- Geographic name
- Methodology
- Supplemental information

Using sub-elements or type qualifiers can extend each of these descriptors. They're optional and repeatable.

History

GILS was developed in the early 1990s as part of a general reorganization of the structure, storage, dissemination and means of access to government information resources. Increasing awareness by the government of the importance of electronic media and networks was another factor in this information restructuring. Reorganization included the creation of the National Information Infrastructure (NII) as

well as the Paperwork Reduction Act, Freedom of Information Act and Records Disposition Act. Of particular importance was the Office of Management of Budget Bulletin 95-1, which originated GILS.

The NII and each of these acts were intended to facilitate access to government information. It was decided a common standard for all government agencies was needed and GILS was considered an ideal tool: an agency-based, network accessible locator of Federal resources. Each agency was subsequently mandated to produce a GILS record for each of their documents and make it available to the public.

Nonetheless, GILS was implemented in different ways by different agencies and with varying levels of success. A report financed by the Defense Department and several other government agencies, "An Evaluation of the Federal Government's Implementation of GILS" (1996), detailed the problems. These included difficulties in coordinating policy, and with management and systems development. There were additional problems with search and retrieval systems, lack of full-text information, inconsistency with hypertext links and even problems in marketing. Most of these difficulties could be traced back to the sheer complexity of GILS, as well as of its origins by government decree. As an example of the latter: there was a lack of initial adequate funding despite the project's mandate from the Office of Management and Budget. Varying degrees of interest in GILS from involved agencies compounded these difficulties, with many agencies considering it more important to develop their own Web sites rather than reconfiguring them with GILS or adding the metadata to new pages. Other problems came from agencies' lack of knowledge concerning Z.30, the international standard on which GILS is based. As a result, according to another report from the Office of Management and Budget, "Launch and Forget: An OMB Watch Report on the Implementation of the U.S. Federal GILS" (1997), by mid-1999, thirty-three government agencies had not posted any GILS records and many others had neglected to update GILS records they had already created.

Of course, sole blame could not be placed on government agencies. For instance, each agency's information was different, both by type and quantity, making it more difficult to apply GILS. Further, government agencies have diverse technological infrastructures and varying levels of skill and readiness among technical people and budgets. Without such networks or technical support, implementation of GILS as mandated would have been very difficult.

Discussion

GILS allows government agencies to organize, identify, locate, describe and retrieve information; further, it offers great depth and complexity in performing the above tasks. GILS has had a difficult history, due to its intricacy, its mandate by government decree and the vast number of widely diverse government agencies involved in its implementation and use. These problems would seem to outweigh GILS' practicality. A note: at the time of this writing, the United States Government information page

about GILS is a “dead” link on the World Wide Web). Still, it must be remembered that GILS is in use and it is hoped that eventually the problems with this metalanguage will be ironed out. Using a simpler metalanguage such as Dublin Core, with similar options and qualities of extensibility, would appear to be a viable, alternative approach for applying metadata to various projects.

Metadata Software Report

TagGen, from Hiawatha Island Software, was chosen as the Dublin Core generator for the Foundations Project. Using TagGen, this metalanguage can be quickly and efficiently inserted into the HTML headers of existing and newly created webpages of Minnesota state agencies, through an editing screen with down-down menus. The screen has been customized for the requirements of the Foundations Project. It includes standards metatags, plus the 15 Dublin Core elements. Due to these menus, no previous cataloging experience in adding meta tags is required.

Another attribute of TagGen is its lack of interference or conflict with previously placed metatags, page design or setup. In addition, TagGen can be mounted on Windows 95 or NT platforms, offers multiple website support and gives the user the ability to do rapid updates of many pages simultaneously.

Other meta-tag-generating software products available are:

- The Metastar product suite, which includes Metastar Data Entry. This allows users to enter metadata, such as qualified Dublin Core, from web browsers into a relational database. This does not seem to be a viable way to create or add metadata to a Web site. Metastar Data Entry can be configured to various data element schemes. It requires a user verification code.
- The Reggie Metadata Editor, which allows creation of new schema files that can be tailored to a particular project's needs. Using these schema, the program can read details of all elements in a set, including characteristics and descriptions. Supporting Dublin Core in five languages, as well as other metalanguages (GILS, ANZLIC, EDNA, AGLS), Reggie allows users to enter and export metadata in a number of syntaxes, to save metadata records to a test repository and to reload these records for editing. Because of these features, however, Reggie appears more complex to use than TagGen.
- The Dublin Core Metadata Template, which lets users describe their web pages on a pre-existing form, then submit it for HTML conversion. After submission, a text editor is used to paste the returned source code into a page's HTML between the <HEAD> and </HEAD> elements. The Metadata Template provides a minimal recommended metadata set with separate keyword elements. It also has an online help guide. While robust in detail, the Template requires cutting and pasting, which makes it more complicated to use and much less efficient for large projects. It also lacks as many qualifiers as TagGen.
- The DC-Dot Dublin Core Generator service, which locates Web pages, Microsoft Office and PowerPoint files and extracts unqualified Dublin Core as either HTML meta tags or RDF. These in turn can be placed into the HTML of a page's source code. The generated metadata can be edited using the form provided and converted to various other formats, including USMARC, SOIF, IAF/ROADS, TEI headers, GILS or RDF. Despite the robust nature of the DC-Dot Generator, there are difficulties in maintaining consistency of keywords.

Description

Resource Description Framework (RDF) is a design for using metadata. Metadata is, simply, data about data; it can be used to organize, identify, locate and describe information on the World Wide Web.

RDF, based on eX tensible Markup Language (XML), was designed to be an infrastructure for metadata schemes that may be useful in different ways to the members of the research community, and as a method for helping encode and bridge these types of metalanguages. It is usually not meant to be a form of metadata applicable by itself; instead, RDF may be considered a foundation that determines, assists and provides consistency to metalanguage(s) used in research-oriented webpages on the World Wide Web.

RDF is intended to help consistently encode metadata, as well as allow its reuse and exchange. The exchange permits semantics, syntax and structure to be bridged and shared among research communities and the various metalanguages they use, both machine- and human-readable.

Exchange may be useful in several areas: resource discovery (enabling better accuracy and expediency in searching), cataloging (increasing detail in content and relationship fields of Web resources), content rating (allowing designation of a particular audience for an Internet site) and intellectual property (attributing ownership to a particular Web page).

Of particular value is how RDF can be used with groups or collections of resources that have similar characteristics. These are called container objects, and RDF defines them three ways:

- 1) as a bag (a list of resources, not in order);
- 2) as a sequence (a list that is in order); or
- 3) as an alternative (a list that may be an option to a single property value, e.g.: a series of Web pages from the same source that are in different languages).

Simple RDF is based on a schema, which includes properties and their associated resources and values. From such a schema, a statement can be developed. This statement may consist of:

- A resource - This term includes any object that can be described on the World Wide Web via a Uniform Resource Locator (URL).
- A property type -This is a value within an object described in the resource field. For example, the term can be used to designate title or author.
- A value - This term names the property within the property type; for example, Fred Ziffel as an author.

Here's an example of a hypothetical statement with Fred Ziffel as the author of a DNR site:

Resource - <http://www.dnr.gov/FredZiffel>

Property type - Author

Value of the property type - Fred Ziffel

History

RDF originated in 1995 under the aegis of the World Wide Web Consortium (W3C). It grew from the Platform for Internet Content Selection (PICS), a device for providing a general method of Web page content ratings (e.g., sexual content, peer review). PICS also served as a non-specialized way of providing such ratings between clients and servers. Nonetheless, eventually PICS was found to have limitations in its depiction of Internet resource content. The W3C created a working group to resolve the situation, the PICS-NG Next Generation (PICSNG), which later became the W3C Resource Description Framework working group. RDF was the result of this group's efforts: inspired by PICS, it is a collaborative, broad-based schema that allowed various metadata communities to have their information made available and used via the Internet.

Discussion

RDF is useful because of these qualities:

- Independence - RDF allows names for its components to be created as needed.
- Interchange - RDF components can easily be converted into its base language, XML, which makes them interchangeable. In other words, it has a simple, useful syntax.
- Scalability - RDF records, with their three-part design may readily be applied to increasingly complex resources.

Emerging XML standards may make RDF unnecessary. XML is capable of incorporating pre-defined metadata, such as Dublin Core. It is likely that in future, XML documents will allow embedded metadata to reside with the appropriate content element, or be linked through definitions. In any case, according to Charlie Morris's article, "Are Search Engines Dead?" dated June 26, 2000, "the Dublin Core/RDF/XML 'stack' seems to be the current state of the art."